Manual de Motif

Creado por Haylem Candelario Bauzá.

2025. Habana Cuba.

Crear una ventana y botón en motif.

```
#include <stdio.h>
#include <stdlib.h>
#include <Xm/Xm.h>
#include <Xm/PushB.h>
#include <Xm/MainW.h>

void exitCallback(Widget widget, XtPointer client_data, XtPointer call_data) {
   exit(0);
}

int main(int argc, char **argv) {
   Widget toplevel, mainWindow, button;
   XtAppContext app;

   // Inicializa el sistema de widgets
   toplevel = XtVaAppInitialize(&app, "SimpleMotif", NULL, 0, &argc, argv, NULL, NULL);

   // Crea la ventana principal
   mainWindow = XmCreateMainWindow(toplevel, "mainWindow", NULL, 0);
   XtManageChild(mainWindow);

   // Crea un botón
   button = XmCreatePushButton(mainWindow, "Exit", NULL, 0);
   XtManageChild(button);

   // Conecta el botón a la función de salida
   XtAddCallback(button, XmNactivateCallback, exitCallback, NULL);

   // Muestra la ventana
   XtRealizeWidget(toplevel);
   XtAppMainLoop(app);

   return 0;
}
```

Compilar con:

```
gcc ejemplo.c -lXm -lXt -lX11
```

Listado de Widgets.

Motif es un conjunto de herramientas de interfaz gráfica que proporciona una variedad de widgets para construir aplicaciones gráficas. A continuación, se presenta un listado de algunos de los widgets más comunes en Motif, junto con una breve descripción de cada uno:

1. TopLevelShell

- **Descripción**: Es la ventana principal de la aplicación. Se utiliza como contenedor para otros widgets.

2. MainWindow

- **Descripción**: Proporciona una estructura básica para la ventana principal de la aplicación, incluyendo áreas para menús, barras de herramientas y contenido.

3. PushButton

- **Descripción**: Un botón que el usuario puede presionar para realizar una acción. Puede contener texto o imágenes.

4. ToggleButton

- **Descripción**: Un botón que puede estar en un estado "activado" o "desactivado". Se utiliza para opciones que pueden ser seleccionadas o deseleccionadas.

5. RadioBox

- **Descripción**: Un grupo de botones de opción (radio buttons) que permite al usuario seleccionar una sola opción de un conjunto.

6. Label

- **Descripción**: Un widget que muestra texto estático. No es interactivo y se utiliza para mostrar información al usuario.

7. Text

- **Descripción**: Un área de texto que permite al usuario ingresar y editar texto. Puede ser de una sola línea o de múltiples líneas.

8. ScrolledWindow

- **Descripción**: Un contenedor que proporciona barras de desplazamiento para su contenido, permitiendo al usuario desplazarse por el contenido que no cabe en la ventana visible.

9. List

- **Descripción**: Un widget que muestra una lista de elementos. Permite seleccionar uno o más elementos de la lista.

10. Menu

- **Descripción**: Un menú desplegable que contiene una lista de opciones. Puede ser utilizado en combinación con menús de barra.

11. MenuButton

- **Descripción**: Un botón que, al ser presionado, despliega un menú. Permite al usuario seleccionar una opción del menú.

12. Dialog

- **Descripción**: Un cuadro de diálogo que se utiliza para mostrar información o solicitar información del usuario. Puede ser modal o no modal.

13. ProgressBar

- **Descripción**: Un widget que muestra el progreso de una operación en curso. Se utiliza comúnmente para indicar que una tarea está en progreso.

14. Separator

- **Descripción**: Un widget que se utiliza para dividir visualmente otros widgets en la interfaz. Puede ser horizontal o vertical.

15. Form

- **Descripción**: Un contenedor que permite organizar otros widgets en una disposición flexible. Los widgets pueden ser colocados en posiciones específicas.

16. Frame

- **Descripción**: Un contenedor que agrupa otros widgets y puede tener un borde. Se utiliza para organizar la interfaz.

17. Scale

- **Descripción**: Un control deslizante que permite al usuario seleccionar un valor de un rango continuo.

18. SpinBox

- **Descripción**: Un widget que permite al usuario seleccionar un valor numérico mediante botones de incremento y decremento.

19. ColorSelector

- **Descripción**: Un widget que permite al usuario seleccionar un color de una paleta.

20. FileSelectionBox

- **Descripción**: Un cuadro de diálogo que permite al usuario seleccionar un archivo del sistema de archivos.

Conclusión

Estos son solo algunos de los widgets disponibles en Motif. Cada widget tiene sus propias propiedades y métodos que se pueden utilizar para personalizar su comportamiento y apariencia. Para obtener más información sobre cada widget, se recomienda consultar la documentación oficial de Motif o los manuales de referencia de la API.

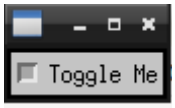
Ejemplos

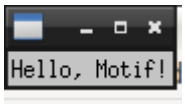
```
#include <stdlib.h>
#include <Xm/Xm.h>
#include <Xm/MainW.h>

int main(int argc, char **argv) {
   Widget toplevel, mainWindow;
   XtAppContext app;

   toplevel = XtVaAppInitialize(&app, "MainWindowExample", NULL, 0, &argc, argv, NULL);
   mainWindow = XmCreateMainWindow(toplevel, "mainWindow", NULL, 0);
   XtManageChild(mainWindow);
   XtRealizeWidget(toplevel);
   XtAppMainLoop(app);

   return 0;
}
```

```
#include <stdlib.h>
#include <stdio.h>
#include <Xm/Xm.h>
#include <Xm/PushB.h>
#include <Xm/MainW.h>

void buttonCallback(Widget widget, XtPointer client_data, XtPointer call_data) {
   printf("Button Pressed!\n");
}

int main(int argc, char **argv) {
   Widget toplevel, mainWindow, button;
   XtAppContext app;

   toplevel = XtVaAppInitialize(&app, "PushButtonExample", NULL, 0, &argc, argv, NULL);
   mainWindow = XmCreateMainWindow(toplevel, "mainWindow", NULL, 0);
   XtManageChild(mainWindow);
```

```
    button = XmCreatePushButton(mainWindow, "Press Me", NULL, 0);
    XtManageChild(button);
    XtAddCallback(button, XmNactivateCallback, buttonCallback, NULL);

    XtRealizeWidget(toplevel);
    XtAppMainLoop(app);

    return 0;
}
```

```
#include <stdio.h>
#include <Xm/Xm.h>
#include <Xm/ToggleB.h>
#include <Xm/MainW.h>

void toggleCallback(Widget widget, XtPointer client_data, XtPointer call_data) {
    Boolean state = XmToggleButtonGetState(widget);
    printf("Toggle Button is %s\n", state ? "ON" : "OFF");
}

int main(int argc, char **argv) {
    Widget toplevel, mainWindow, toggleButton;
    XtAppContext app;

    toplevel = XtVaAppInitialize(&app, "ToggleButtonExample", NULL, 0, &argc, argv, NULL);
    mainWindow = XmCreateMainWindow(toplevel, "mainWindow", NULL, 0);
    XtManageChild(mainWindow);

    toggleButton = XmCreateToggleButton(mainWindow, "Toggle Me", NULL, 0);
    XtManageChild(toggleButton);
    XtAddCallback(toggleButton, XmNvalueChangedCallback, toggleCallback, NULL);

    XtRealizeWidget(toplevel);
    XtAppMainLoop(app);

    return 0;
}
```




```
#include <Xm/Xm.h>
#include <Xm/Label.h>
#include <Xm/MainW.h>
```

```
int main(int argc, char **argv) {
    Widget toplevel, mainWindow, label;
    XtAppContext app;

    toplevel = XtVaAppInitialize(&app, "LabelExample", NULL, 0, &argc, argv, NULL);
    mainWindow = XmCreateMainWindow(toplevel, "mainWindow", NULL, 0);
    XtManageChild(mainWindow);

    label = XmCreateLabel(mainWindow, "Hello, Motif!", NULL, 0);
    XtManageChild(label);

    XtRealizeWidget(toplevel);
    XtAppMainLoop(app);

    return 0;
}
```